

Party Example

Party

```
public class Character {  
    private int battlesWon = 0;  
    private int expPoints = 0;  
  
    public Character() {}  
    public void winBattle(int xp) {  
        this.battlesWon++;  
        this.expPoints += xp;  
    }  
}
```

```
public class Party {  
    private ArrayList<Character> members;  
    private int battlesWon = 0;  
  
    public Party() {  
        this.members = new ArrayList<>();  
    }  
    public void addCharacter(Character member) {  
        this.members.add(member);  
    }  
    public void winBattle(int xp) {  
        this.battlesWon++;  
        for (int x=0; x < this.members.size(); x++) {  
            this.members.get(x).winBattle(xp);  
        }  
    }  
    public static void main(String[] args) {  
        Character hero = new Character();  
        hero = new Character();  
        Character fighter = new Character();  
        hero.winBattle(10);  
        Party party = new Party();  
        party.addCharacter(hero);  
        party.addCharacter(fighter);  
        party.winBattle(20);  
    }  
}
```

- Let's look at another example using objects
- We'll highlight 2 new concepts with this code:
- Composition
- Garbage Collection

Composition

- Instance variables of objects can store references to other objects
- The Party class is composed of an ArrayList of Characters

```
public class Character {  
    private int battlesWon = 0;  
    private int expPoints = 0;  
  
    public Character() {}  
    public void winBattle(int xp) {  
        this.battlesWon++;  
        this.expPoints += xp;  
    }  
}
```

```
public class Party {  
    private ArrayList<Character> members;  
    private int battlesWon = 0;  
  
    public Party() {  
        this.members = new ArrayList<>();  
    }  
    public void addCharacter(Character member) {  
        this.members.add(member);  
    }  
    public void winBattle(int xp) {  
        this.battlesWon++;  
        for (int x=0; x < this.members.size(); x++) {  
            this.members.get(x).winBattle(xp);  
        }  
    }  
    public static void main(String[] args) {  
        Character hero = new Character();  
        hero = new Character();  
        Character fighter = new Character();  
        hero.winBattle(10);  
        Party party = new Party();  
        party.addCharacter(hero);  
        party.addCharacter(fighter);  
        party.winBattle(20);  
    }  
}
```


Garbage Collection

- hero is assigned a reference to an instance/object of type Character
- hero is then reassigned to a new reference
- We no longer have a reference to the first Character object in memory
- Since we cannot access this object, it will be removed from memory

```
public class Character {  
    private int battlesWon = 0;  
    private int expPoints = 0;  
  
    public Character() {}  
    public void winBattle(int xp) {  
        this.battlesWon++;  
        this.expPoints += xp;  
    }  
}
```

```
public class Party {  
    private ArrayList<Character> members;  
    private int battlesWon = 0;  
  
    public Party() {  
        this.members = new ArrayList<>();  
    }  
    public void addCharacter(Character member) {  
        this.members.add(member);  
    }  
    public void winBattle(int xp) {  
        this.battlesWon++;  
        for (int x=0; x < this.members.size(); x++) {  
            this.members.get(x).winBattle(xp);  
        }  
    }  
    public static void main(String[] args) {  
        Character hero = new Character();  
        hero = new Character();  
        Character fighter = new Character();  
        hero.winBattle(10);  
        Party party = new Party();  
        party.addCharacter(hero);  
        party.addCharacter(fighter);  
        party.winBattle(20);  
    }  
}
```

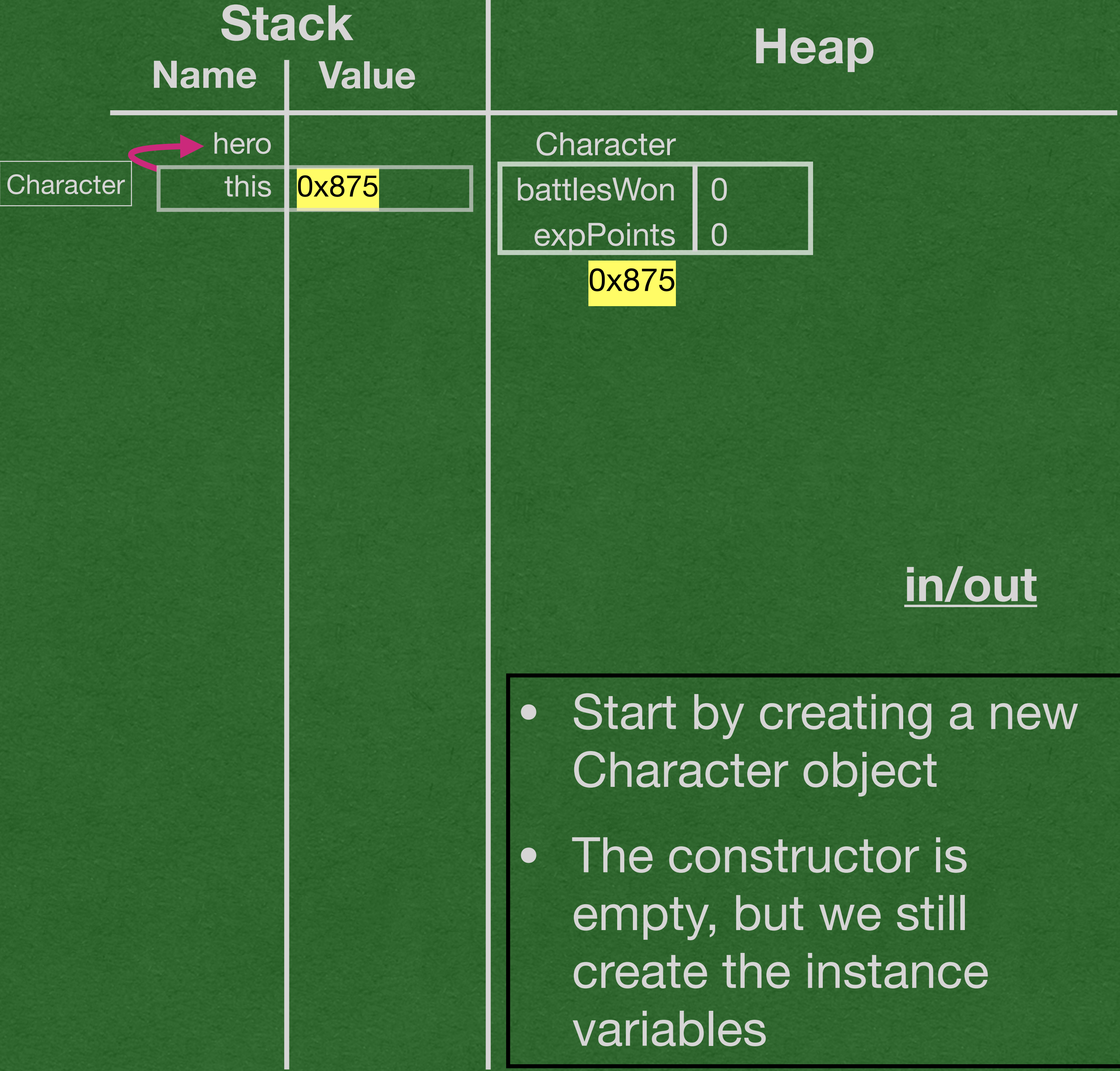

Memory Diagram


```
public class Character {
    private int battlesWon = 0;
    private int expPoints = 0;

    public Character() {}
    public void winBattle(int xp) {
        this.battlesWon++;
        this.expPoints += xp;
    }
}
```

```
public class Party {
    private ArrayList<Character> members;
    private int battlesWon = 0;

    public Party() {
        this.members = new ArrayList<>();
    }
    public void addCharacter(Character member) {
        this.members.add(member);
    }
    public void winBattle(int xp) {
        this.battlesWon++;
        for (int x=0; x < this.members.size(); x++) {
            this.members.get(x).winBattle(xp);
        }
    }
    public static void main(String[] args) {
        Character hero = new Character();
        hero = new Character();
        Character fighter = new Character();
        hero.winBattle(10);
        Party party = new Party();
        party.addCharacter(hero);
        party.addCharacter(fighter);
        party.winBattle(20);
    }
}
```

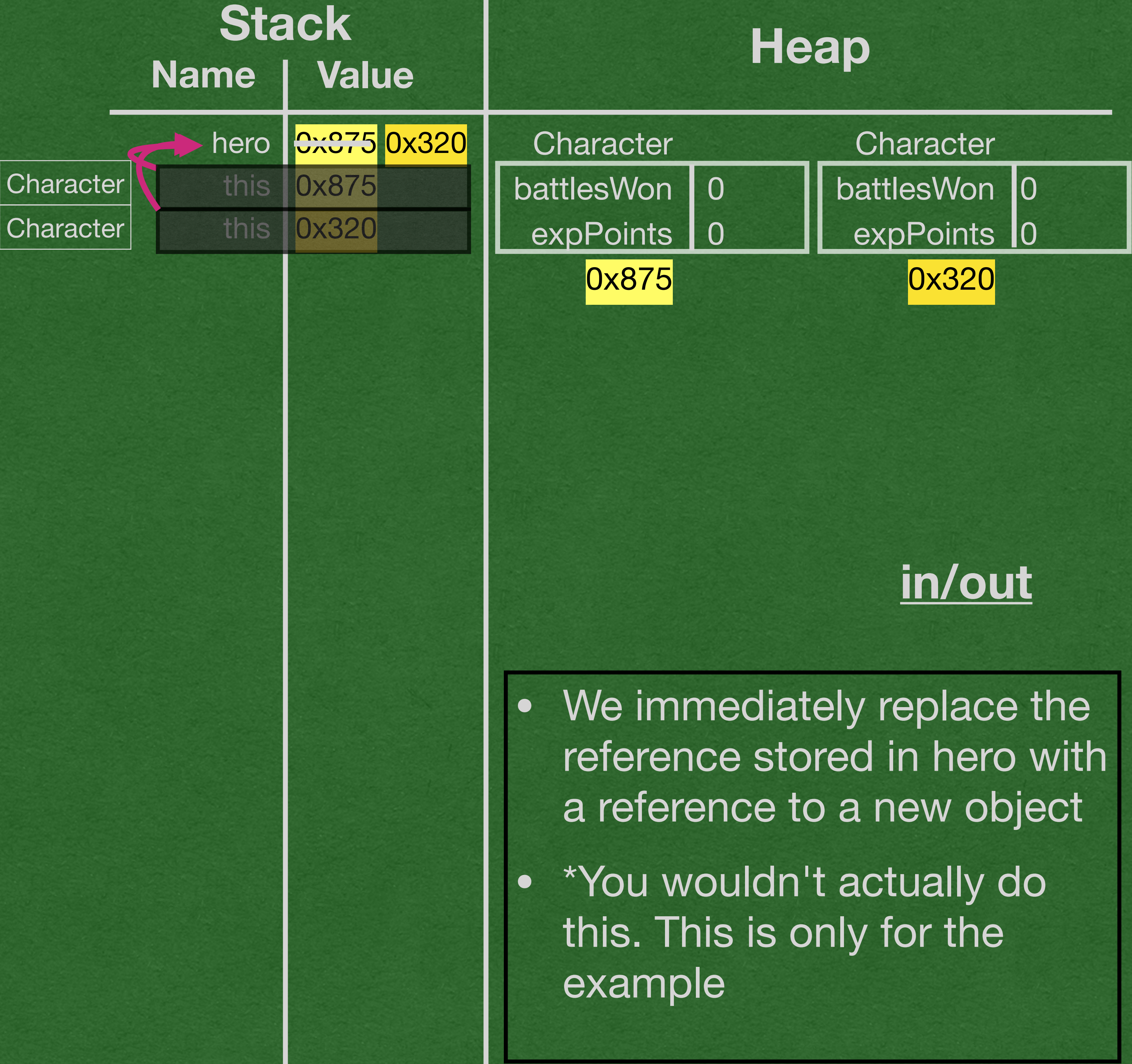
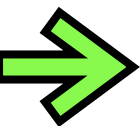



```
public class Character {
    private int battlesWon = 0;
    private int expPoints = 0;

    public Character() {}
    public void winBattle(int xp) {
        this.battlesWon++;
        this.expPoints += xp;
    }
}
```

```
public class Party {
    private ArrayList<Character> members;
    private int battlesWon = 0;

    public Party() {
        this.members = new ArrayList<>();
    }
    public void addCharacter(Character member) {
        this.members.add(member);
    }
    public void winBattle(int xp) {
        this.battlesWon++;
        for (int x=0; x < this.members.size(); x++) {
            this.members.get(x).winBattle(xp);
        }
    }
    public static void main(String[] args) {
        Character hero = new Character();
        hero = new Character();
        Character fighter = new Character();
        hero.winBattle(10);
        Party party = new Party();
        party.addCharacter(hero);
        party.addCharacter(fighter);
        party.winBattle(20);
    }
}
```

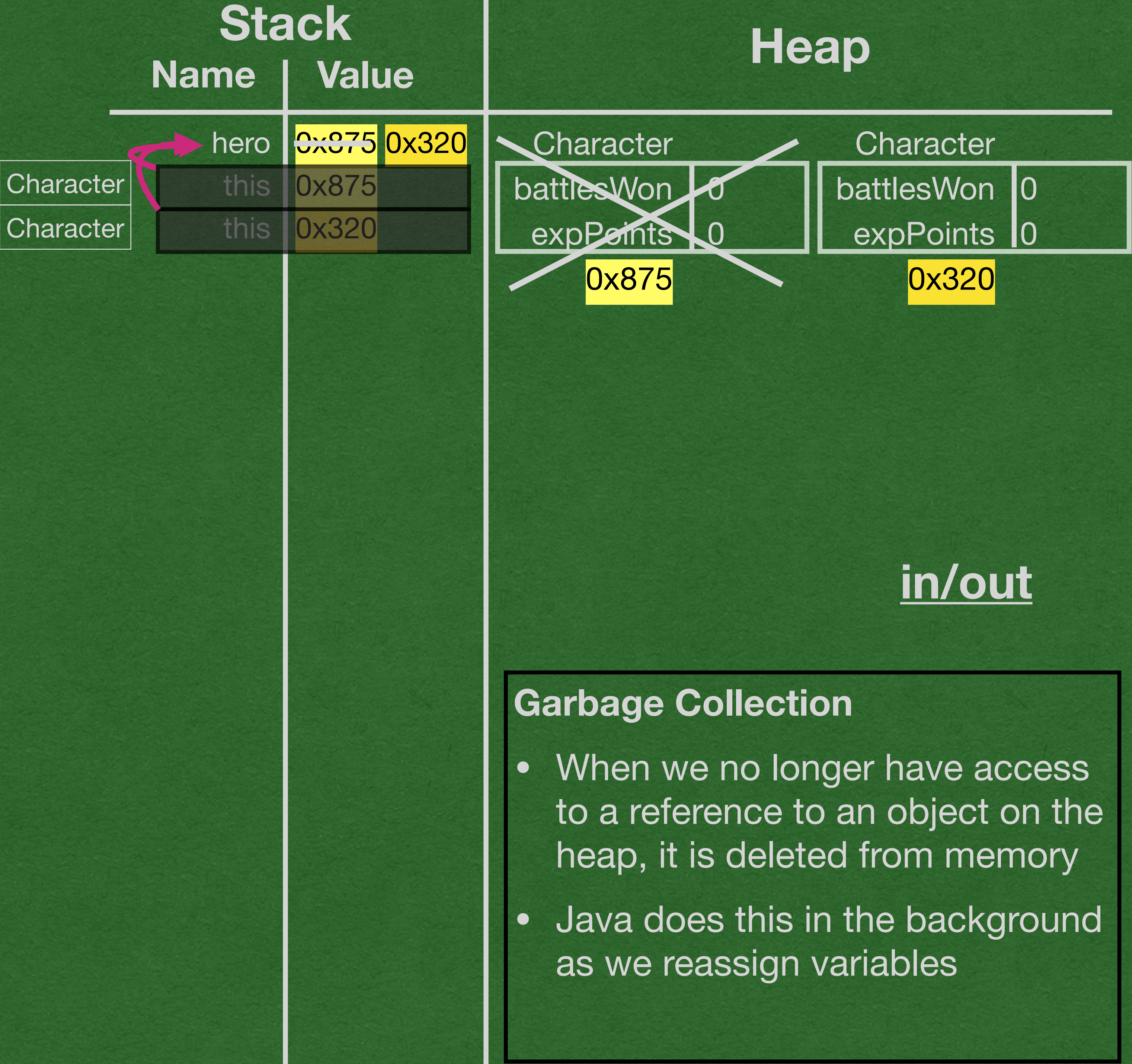
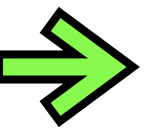



```
public class Character {
    private int battlesWon = 0;
    private int expPoints = 0;

    public Character() {}
    public void winBattle(int xp) {
        this.battlesWon++;
        this.expPoints += xp;
    }
}
```

```
public class Party {
    private ArrayList<Character> members;
    private int battlesWon = 0;

    public Party() {
        this.members = new ArrayList<>();
    }
    public void addCharacter(Character member) {
        this.members.add(member);
    }
    public void winBattle(int xp) {
        this.battlesWon++;
        for (int x=0; x < this.members.size(); x++) {
            this.members.get(x).winBattle(xp);
        }
    }
    public static void main(String[] args) {
        Character hero = new Character();
        hero = new Character();
        Character fighter = new Character();
        hero.winBattle(10);
        Party party = new Party();
        party.addCharacter(hero);
        party.addCharacter(fighter);
        party.winBattle(20);
    }
}
```

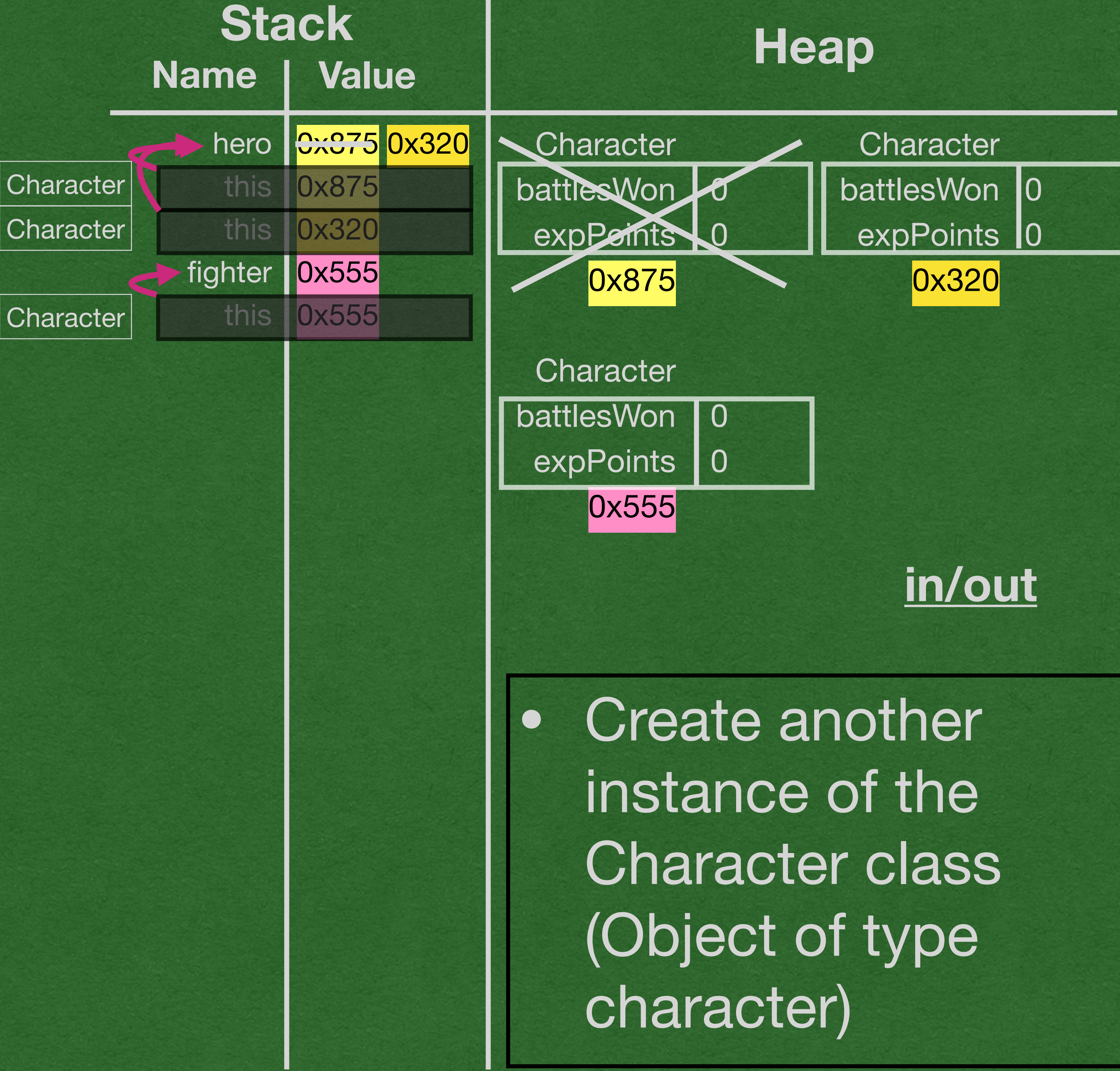
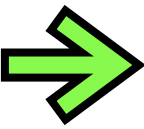



```
public class Character {
    private int battlesWon = 0;
    private int expPoints = 0;

    public Character() {}
    public void winBattle(int xp) {
        this.battlesWon++;
        this.expPoints += xp;
    }
}
```

```
public class Party {
    private ArrayList<Character> members;
    private int battlesWon = 0;

    public Party() {
        this.members = new ArrayList<>();
    }
    public void addCharacter(Character member) {
        this.members.add(member);
    }
    public void winBattle(int xp) {
        this.battlesWon++;
        for (int x=0; x < this.members.size(); x++) {
            this.members.get(x).winBattle(xp);
        }
    }
    public static void main(String[] args) {
        Character hero = new Character();
        hero = new Character();
        Character fighter = new Character();
        hero.winBattle(10);
        Party party = new Party();
        party.addCharacter(hero);
        party.addCharacter(fighter);
        party.winBattle(20);
    }
}
```

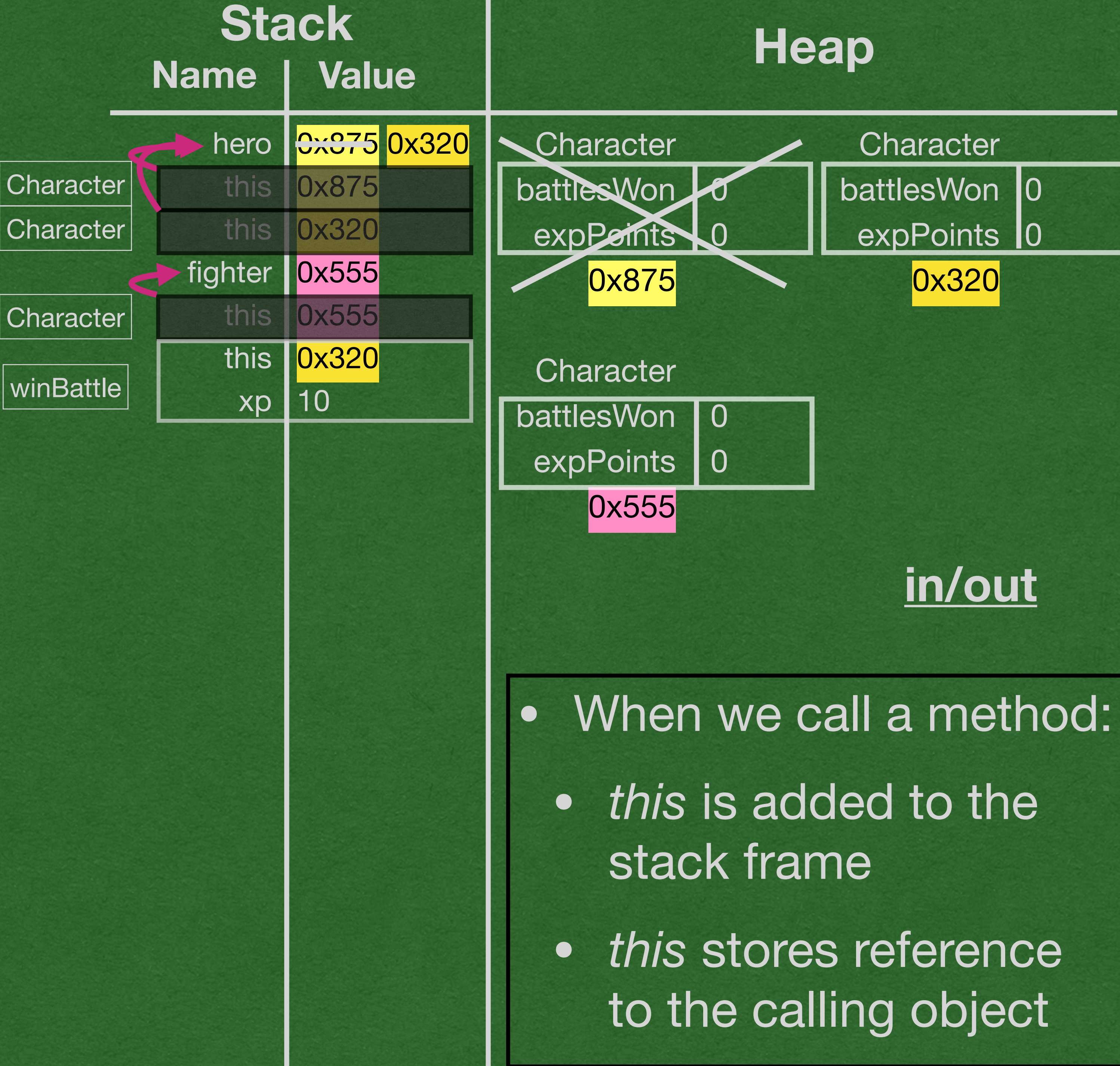



```
public class Character {
    private int battlesWon = 0;
    private int expPoints = 0;

    public Character() {}
    public void winBattle(int xp) {
        this.battlesWon++;
        this.expPoints += xp;
    }
}
```

```
public class Party {
    private ArrayList<Character> members;
    private int battlesWon = 0;

    public Party() {
        this.members = new ArrayList<>();
    }
    public void addCharacter(Character member) {
        this.members.add(member);
    }
    public void winBattle(int xp) {
        this.battlesWon++;
        for (int x=0; x < this.members.size(); x++) {
            this.members.get(x).winBattle(xp);
        }
    }
    public static void main(String[] args) {
        Character hero = new Character();
        hero = new Character();
        Character fighter = new Character();
        hero.winBattle(10);
        Party party = new Party();
        party.addCharacter(hero);
        party.addCharacter(fighter);
        party.winBattle(20);
    }
}
```

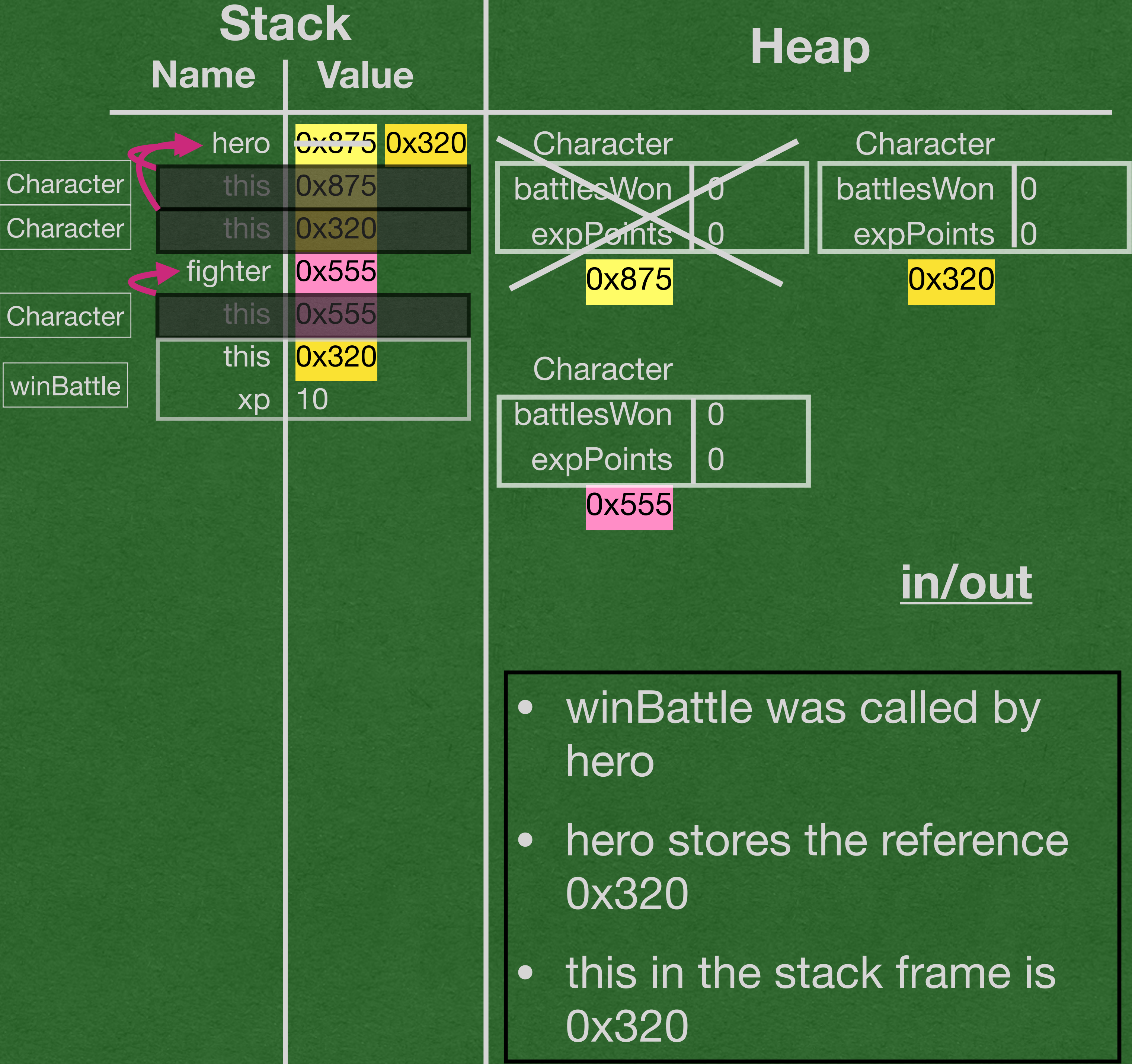



```
public class Character {
    private int battlesWon = 0;
    private int expPoints = 0;

    public Character() {}
    public void winBattle(int xp) {
        this.battlesWon++;
        this.expPoints += xp;
    }
}
```

```
public class Party {
    private ArrayList<Character> members;
    private int battlesWon = 0;

    public Party() {
        this.members = new ArrayList<>();
    }
    public void addCharacter(Character member) {
        this.members.add(member);
    }
    public void winBattle(int xp) {
        this.battlesWon++;
        for (int x=0; x < this.members.size(); x++) {
            this.members.get(x).winBattle(xp);
        }
    }
    public static void main(String[] args) {
        Character hero = new Character();
        hero = new Character();
        Character fighter = new Character();
        hero.winBattle(10);
        Party party = new Party();
        party.addCharacter(hero);
        party.addCharacter(fighter);
        party.winBattle(20);
    }
}
```

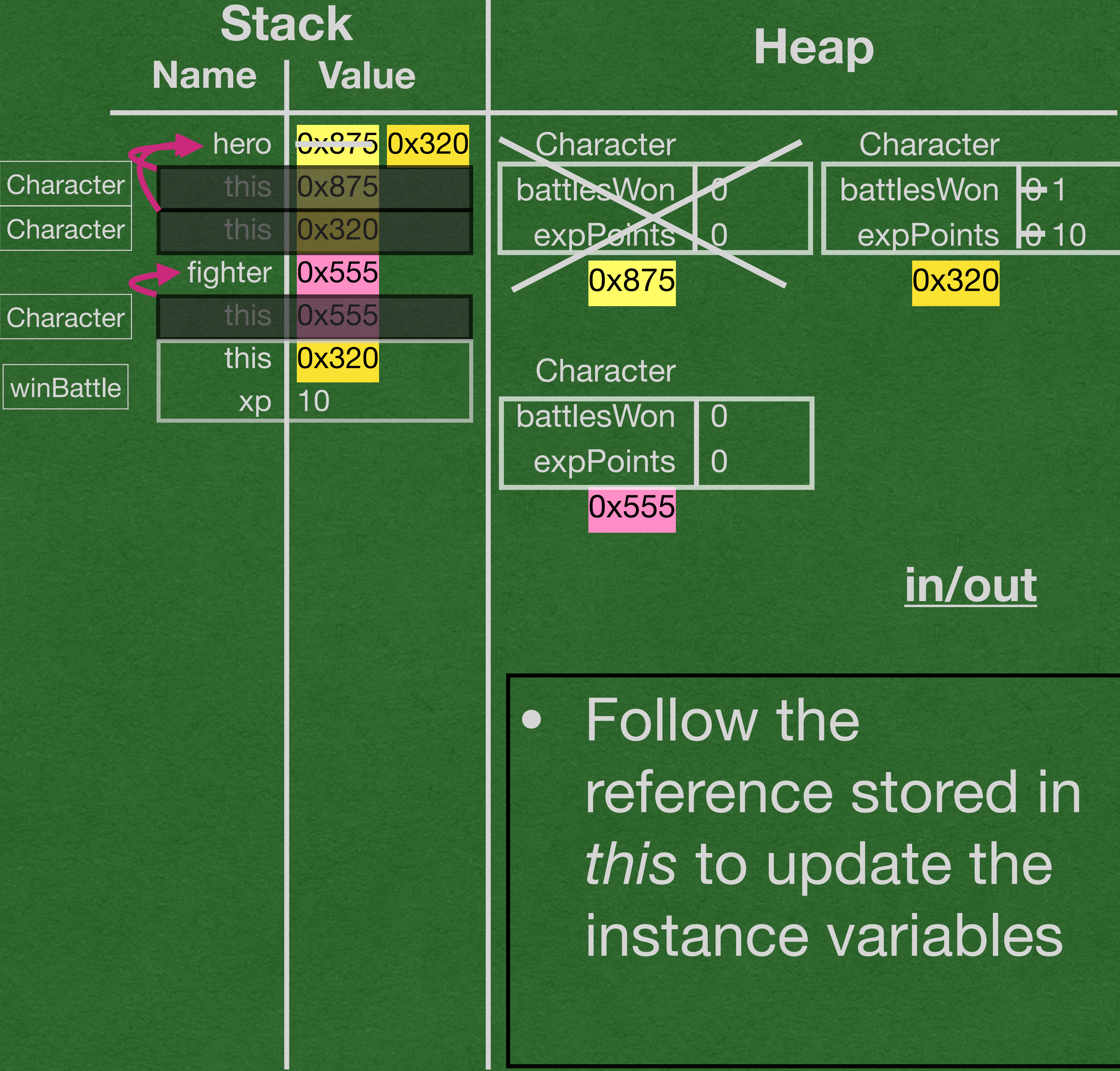



```
public class Character {
    private int battlesWon = 0;
    private int expPoints = 0;

    public Character() {}
    public void winBattle(int xp) {
        this.battlesWon++;
        this.expPoints += xp;
    }
}
```

```
public class Party {
    private ArrayList<Character> members;
    private int battlesWon = 0;

    public Party() {
        this.members = new ArrayList<>();
    }
    public void addCharacter(Character member) {
        this.members.add(member);
    }
    public void winBattle(int xp) {
        this.battlesWon++;
        for (int x=0; x < this.members.size(); x++) {
            this.members.get(x).winBattle(xp);
        }
    }
    public static void main(String[] args) {
        Character hero = new Character();
        hero = new Character();
        Character fighter = new Character();
        hero.winBattle(10);
        Party party = new Party();
        party.addCharacter(hero);
        party.addCharacter(fighter);
        party.winBattle(20);
    }
}
```

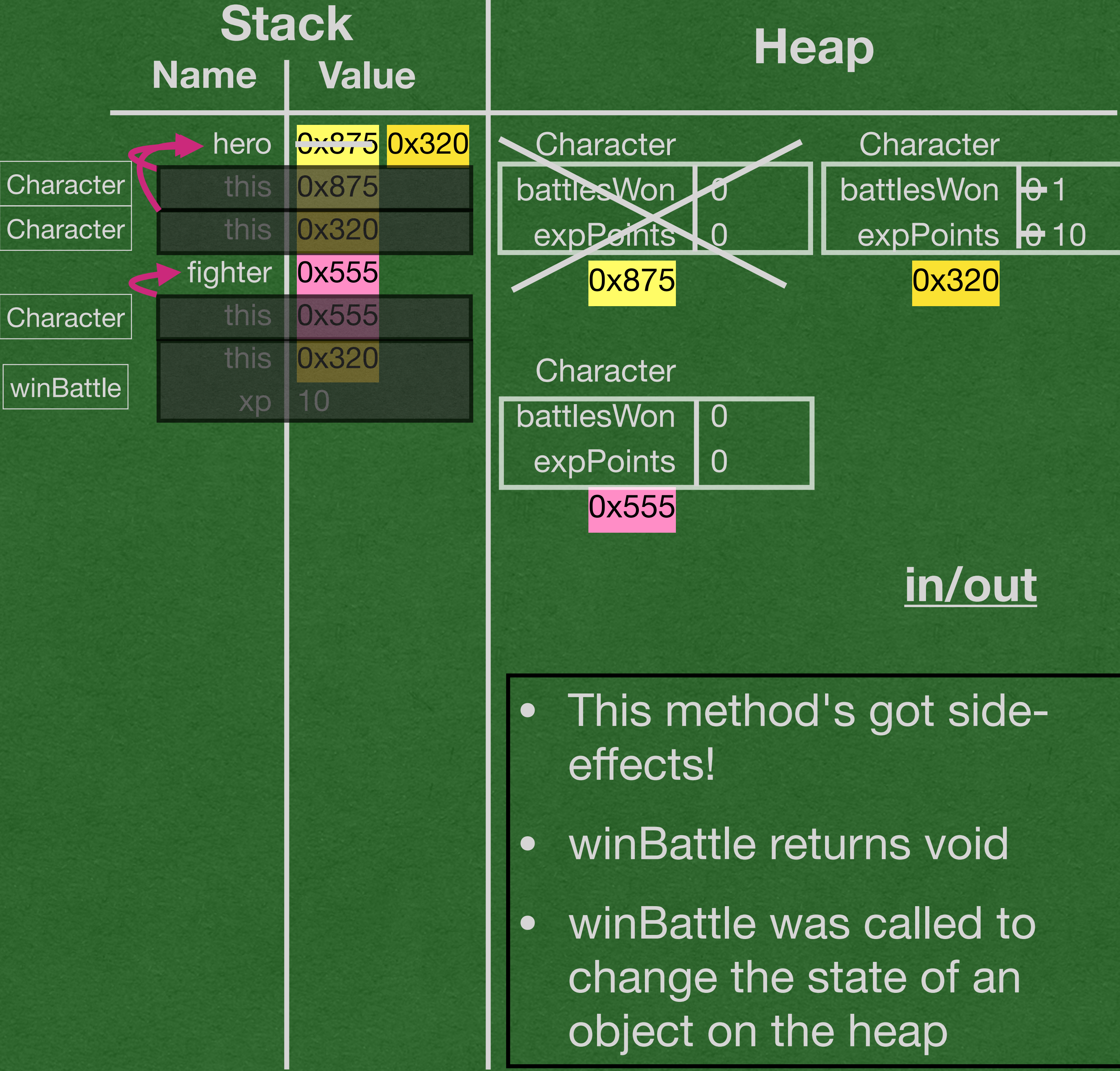



```
public class Character {
    private int battlesWon = 0;
    private int expPoints = 0;

    public Character() {}
    public void winBattle(int xp) {
        this.battlesWon++;
        this.expPoints += xp;
    }
}
```

```
public class Party {
    private ArrayList<Character> members;
    private int battlesWon = 0;

    public Party() {
        this.members = new ArrayList<>();
    }
    public void addCharacter(Character member) {
        this.members.add(member);
    }
    public void winBattle(int xp) {
        this.battlesWon++;
        for (int x=0; x < this.members.size(); x++) {
            this.members.get(x).winBattle(xp);
        }
    }
    public static void main(String[] args) {
        Character hero = new Character();
        hero = new Character();
        Character fighter = new Character();
        hero.winBattle(10);
        Party party = new Party();
        party.addCharacter(hero);
        party.addCharacter(fighter);
        party.winBattle(20);
    }
}
```

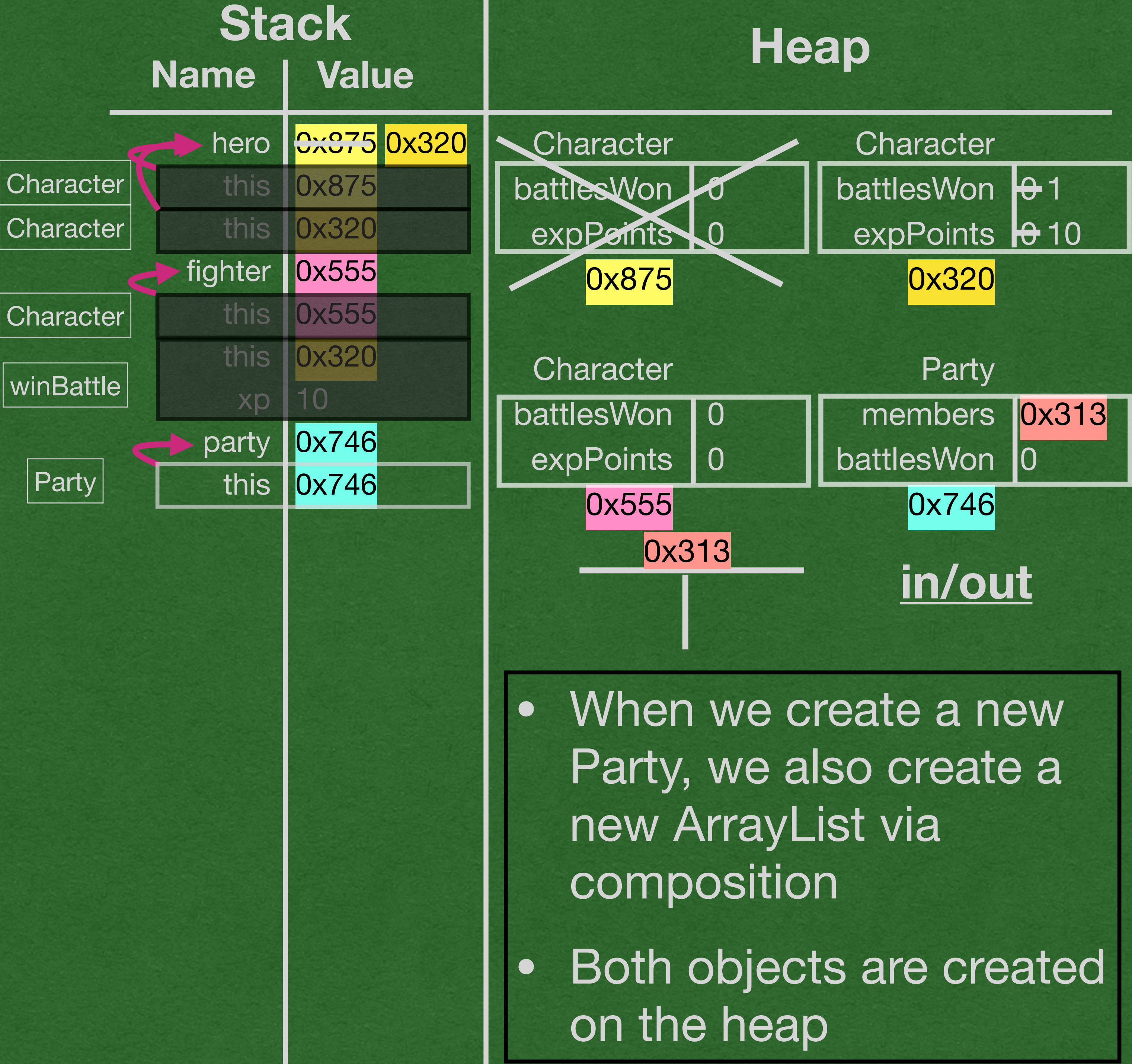



```
public class Character {
    private int battlesWon = 0;
    private int expPoints = 0;

    public Character() {}
    public void winBattle(int xp) {
        this.battlesWon++;
        this.expPoints += xp;
    }
}
```

```
public class Party {
    private ArrayList<Character> members;
    private int battlesWon = 0;

    public Party() {
        this.members = new ArrayList<>();
    }
    public void addCharacter(Character member) {
        this.members.add(member);
    }
    public void winBattle(int xp) {
        this.battlesWon++;
        for (int x=0; x < this.members.size(); x++) {
            this.members.get(x).winBattle(xp);
        }
    }
    public static void main(String[] args) {
        Character hero = new Character();
        hero = new Character();
        Character fighter = new Character();
        hero.winBattle(10);
        Party party = new Party();
        party.addCharacter(hero);
        party.addCharacter(fighter);
        party.winBattle(20);
    }
}
```



members

0x313

battlesWon

0

0x746

in/out

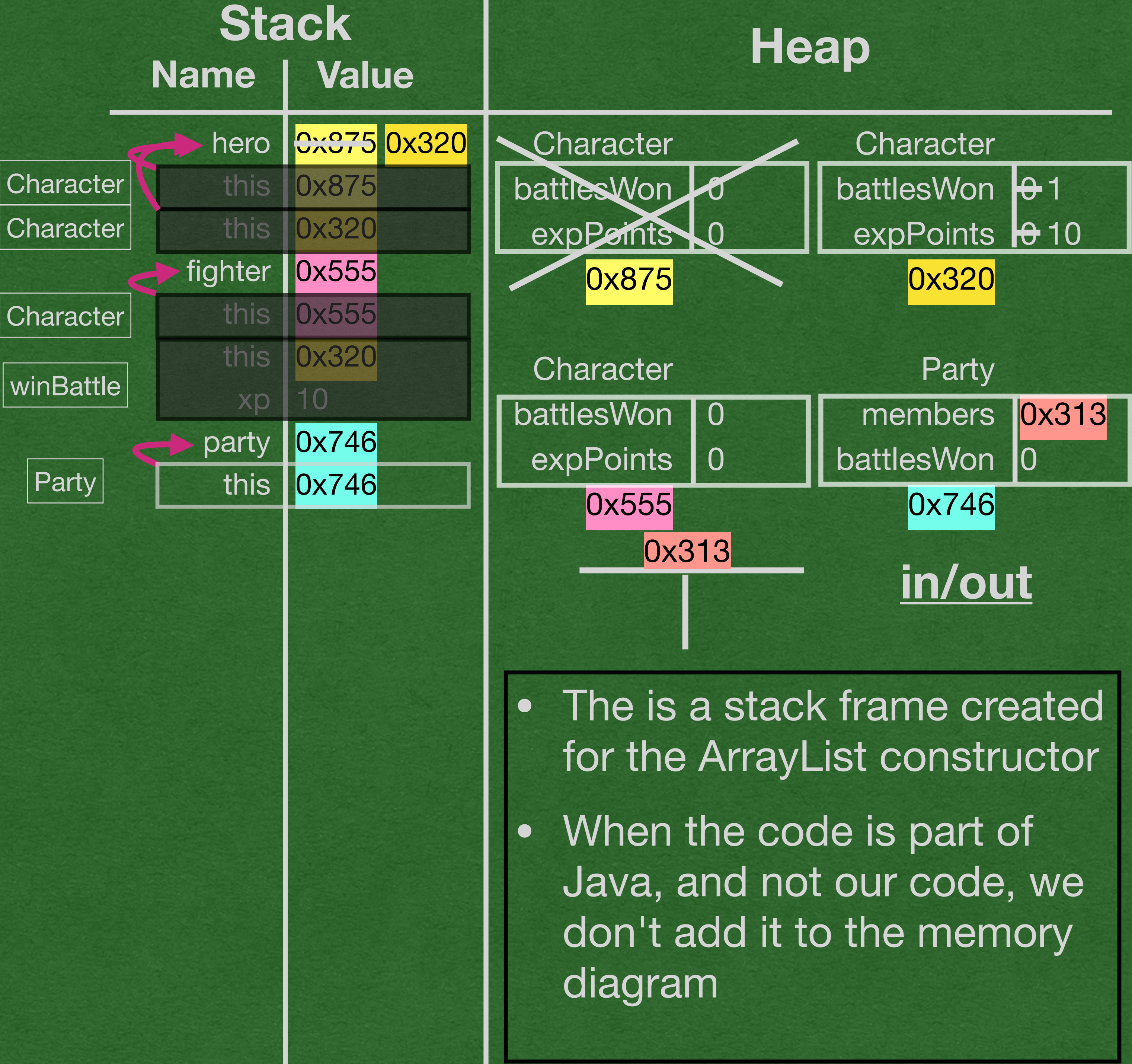
- When we create a new Party, we also create a new ArrayList via composition
- Both objects are created on the heap


```
public class Character {
    private int battlesWon = 0;
    private int expPoints = 0;

    public Character() {}
    public void winBattle(int xp) {
        this.battlesWon++;
        this.expPoints += xp;
    }
}
```

```
public class Party {
    private ArrayList<Character> members;
    private int battlesWon = 0;

    public Party() {
        this.members = new ArrayList<>();
    }
    public void addCharacter(Character member) {
        this.members.add(member);
    }
    public void winBattle(int xp) {
        this.battlesWon++;
        for (int x=0; x < this.members.size(); x++) {
            this.members.get(x).winBattle(xp);
        }
    }
    public static void main(String[] args) {
        Character hero = new Character();
        hero = new Character();
        Character fighter = new Character();
        hero.winBattle(10);
        Party party = new Party();
        party.addCharacter(hero);
        party.addCharacter(fighter);
        party.winBattle(20);
    }
}
```

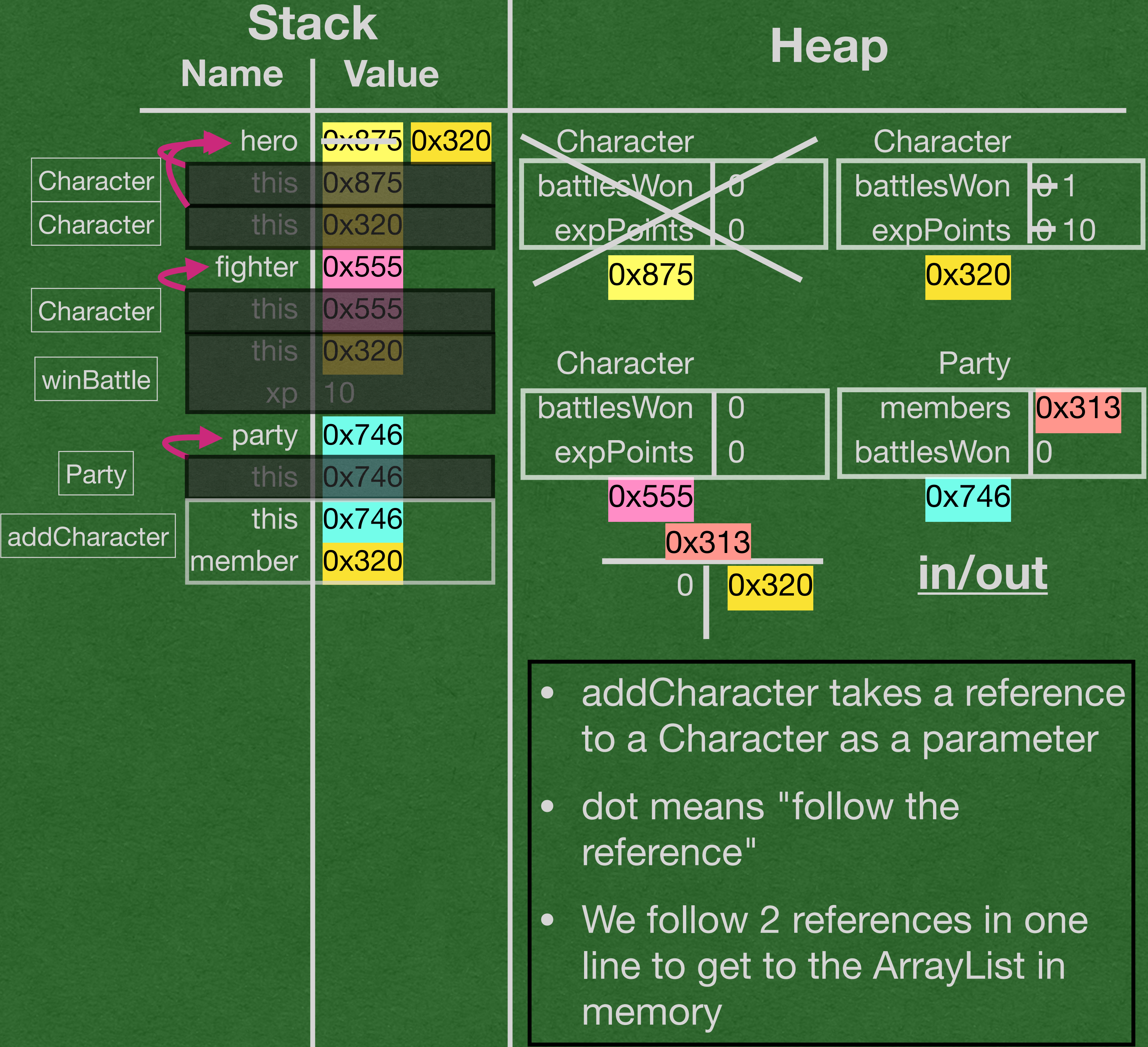



```
public class Character {
    private int battlesWon = 0;
    private int expPoints = 0;

    public Character() {}
    public void winBattle(int xp) {
        this.battlesWon++;
        this.expPoints += xp;
    }
}
```

```
public class Party {
    private ArrayList<Character> members;
    private int battlesWon = 0;

    public Party() {
        this.members = new ArrayList<>();
    }
    public void addCharacter(Character member) {
        this.members.add(member);
    }
    public void winBattle(int xp) {
        this.battlesWon++;
        for (int x=0; x < this.members.size(); x++) {
            this.members.get(x).winBattle(xp);
        }
    }
    public static void main(String[] args) {
        Character hero = new Character();
        hero = new Character();
        Character fighter = new Character();
        hero.winBattle(10);
        Party party = new Party();
        party.addCharacter(hero);
        party.addCharacter(fighter);
        party.winBattle(20);
    }
}
```

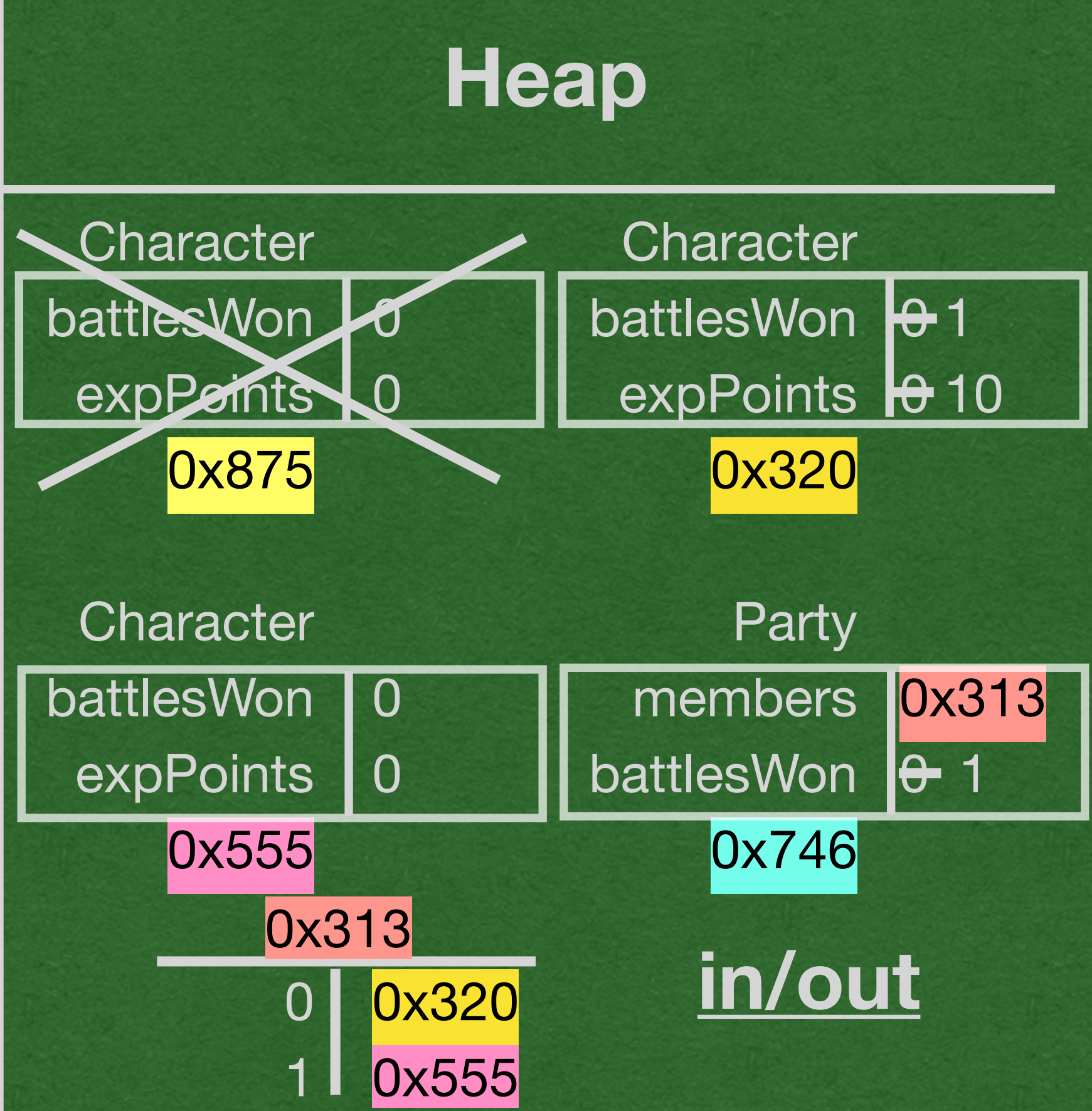
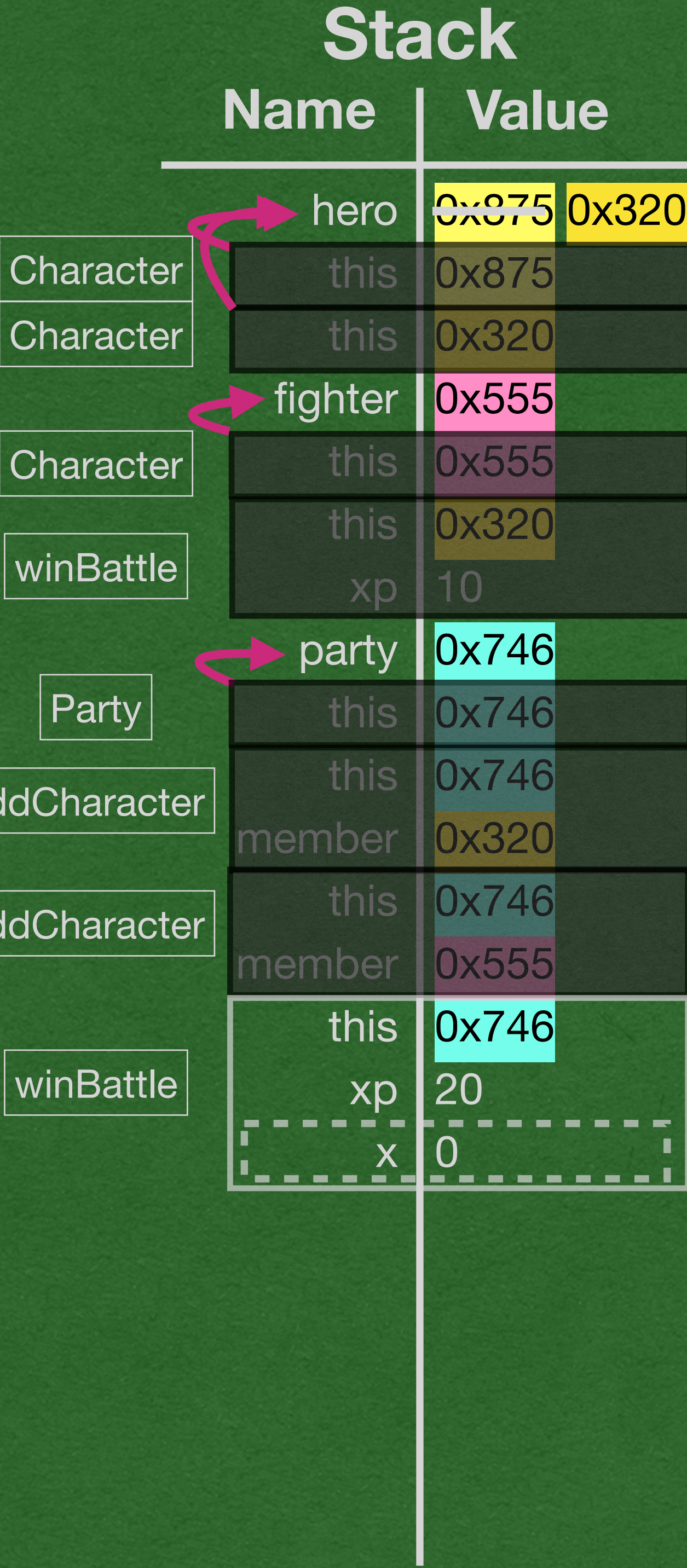



```
public class Character {
    private int battlesWon = 0;
    private int expPoints = 0;

    public Character() {}
    public void winBattle(int xp) {
        this.battlesWon++;
        this.expPoints += xp;
    }
}
```

```
public class Party {
    private ArrayList<Character> members;
    private int battlesWon = 0;

    public Party() {
        this.members = new ArrayList<>();
    }
    public void addCharacter(Character member) {
        this.members.add(member);
    }
    public void winBattle(int xp) {
        this.battlesWon++;
        for (int x=0; x < this.members.size(); x++) {
            this.members.get(x).winBattle(xp);
        }
    }
    public static void main(String[] args) {
        Character hero = new Character();
        hero = new Character();
        Character fighter = new Character();
        hero.winBattle(10);
        Party party = new Party();
        party.addCharacter(hero);
        party.addCharacter(fighter);
        party.winBattle(20);
    }
}
```



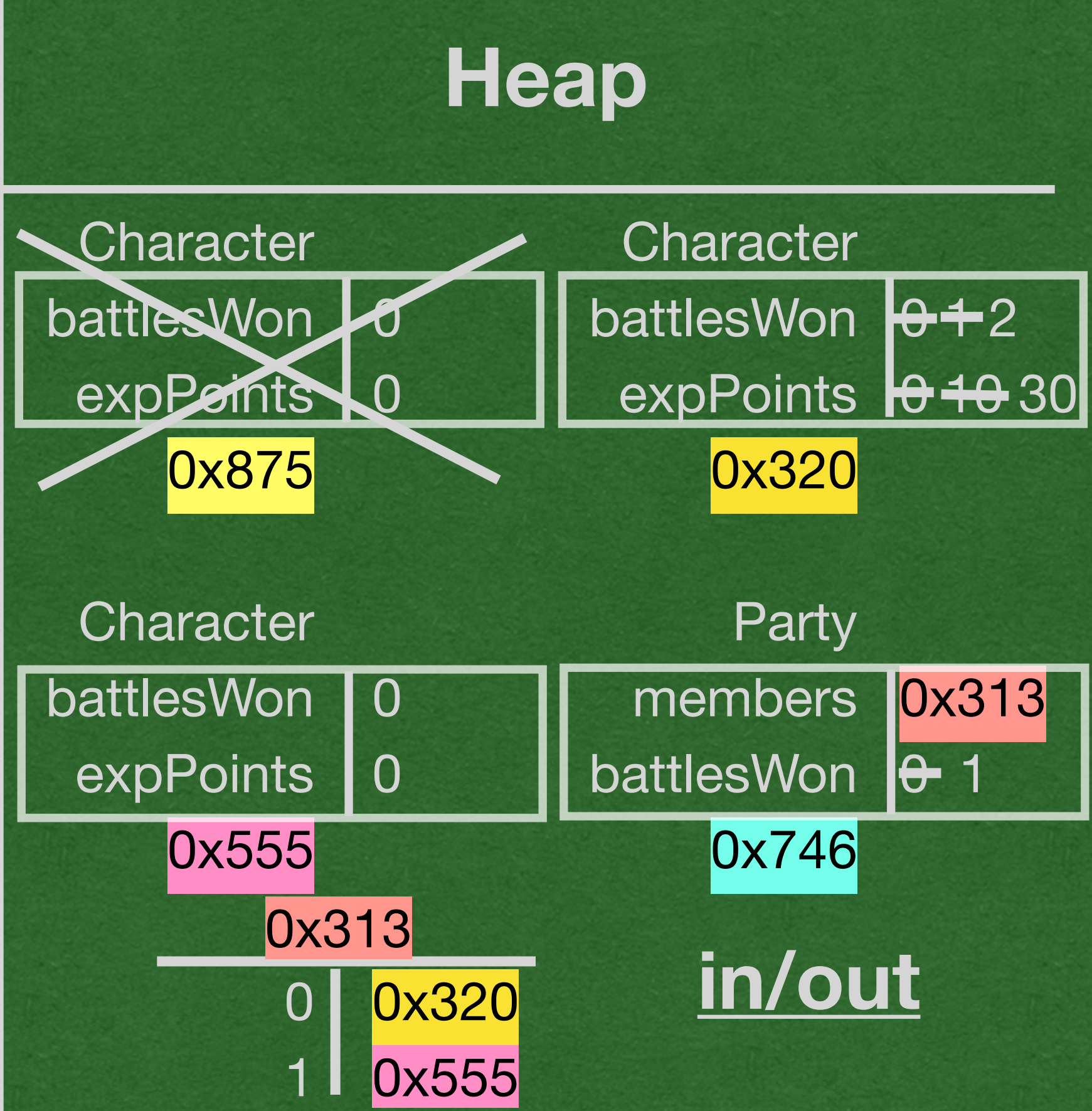
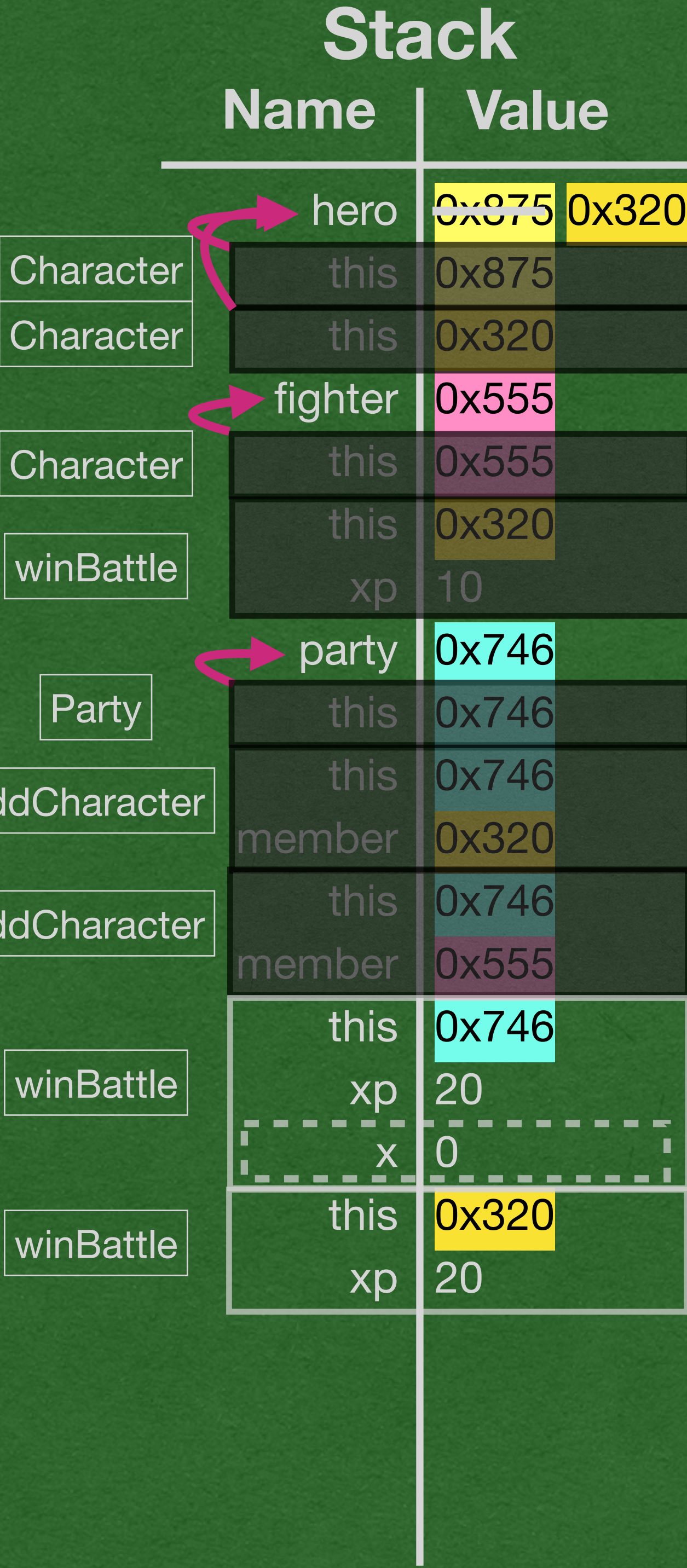
- Check out this line!
- dot means follow the reference
- We'll jump around in memory 3 times to reach the Player at 0x320


```
public class Character {
    private int battlesWon = 0;
    private int expPoints = 0;

    public Character() {}
    public void winBattle(int xp) {
        this.battlesWon++;
        this.expPoints += xp;
    }
}
```

```
public class Party {
    private ArrayList<Character> members;
    private int battlesWon = 0;

    public Party() {
        this.members = new ArrayList<>();
    }
    public void addCharacter(Character member) {
        this.members.add(member);
    }
    public void winBattle(int xp) {
        this.battlesWon++;
        for (int x=0; x < this.members.size(); x++) {
            this.members.get(x).winBattle(xp);
        }
    }
    public static void main(String[] args) {
        Character hero = new Character();
        hero = new Character();
        Character fighter = new Character();
        hero.winBattle(10);
        Party party = new Party();
        party.addCharacter(hero);
        party.addCharacter(fighter);
        party.winBattle(20);
    }
}
```



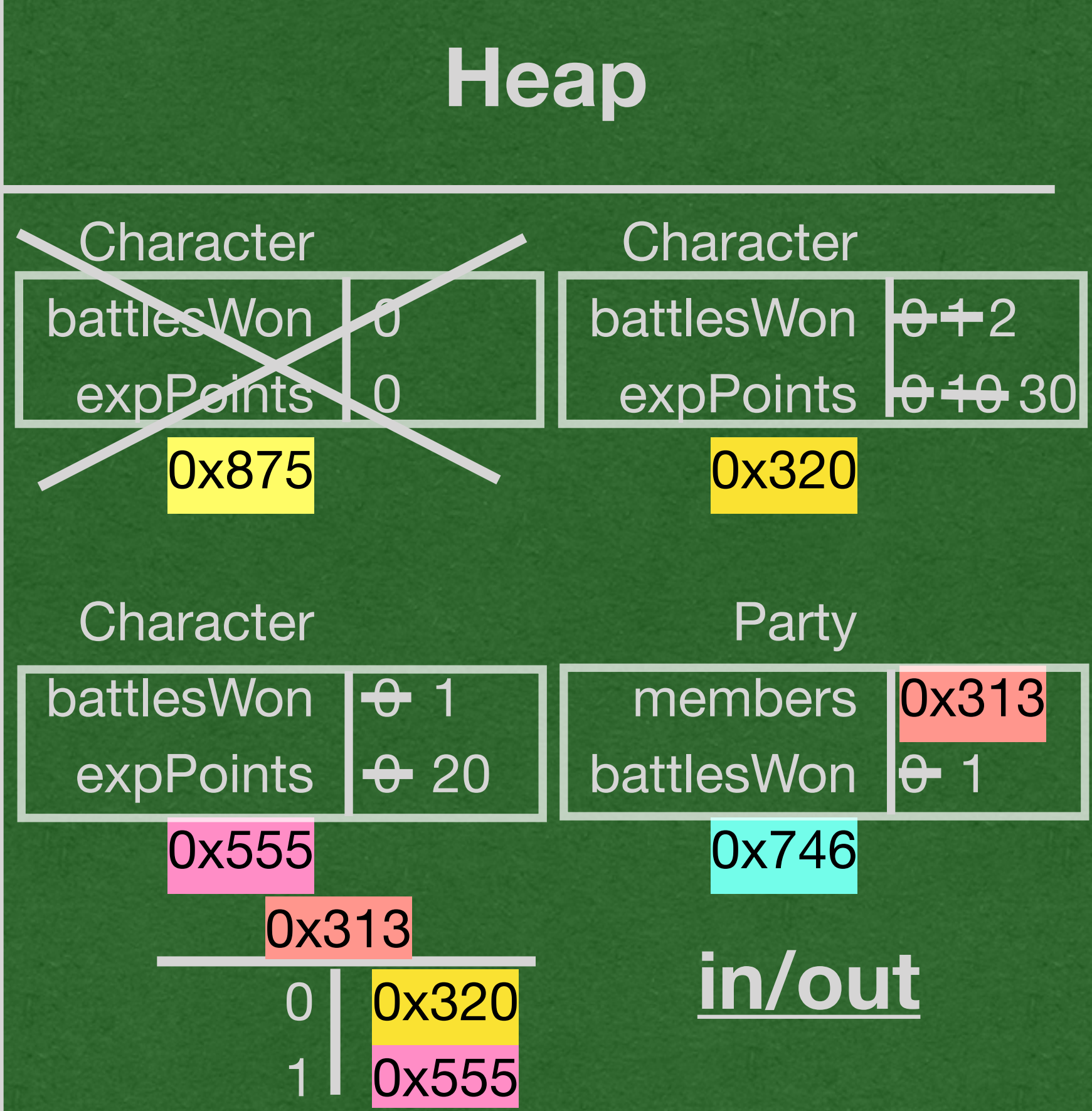
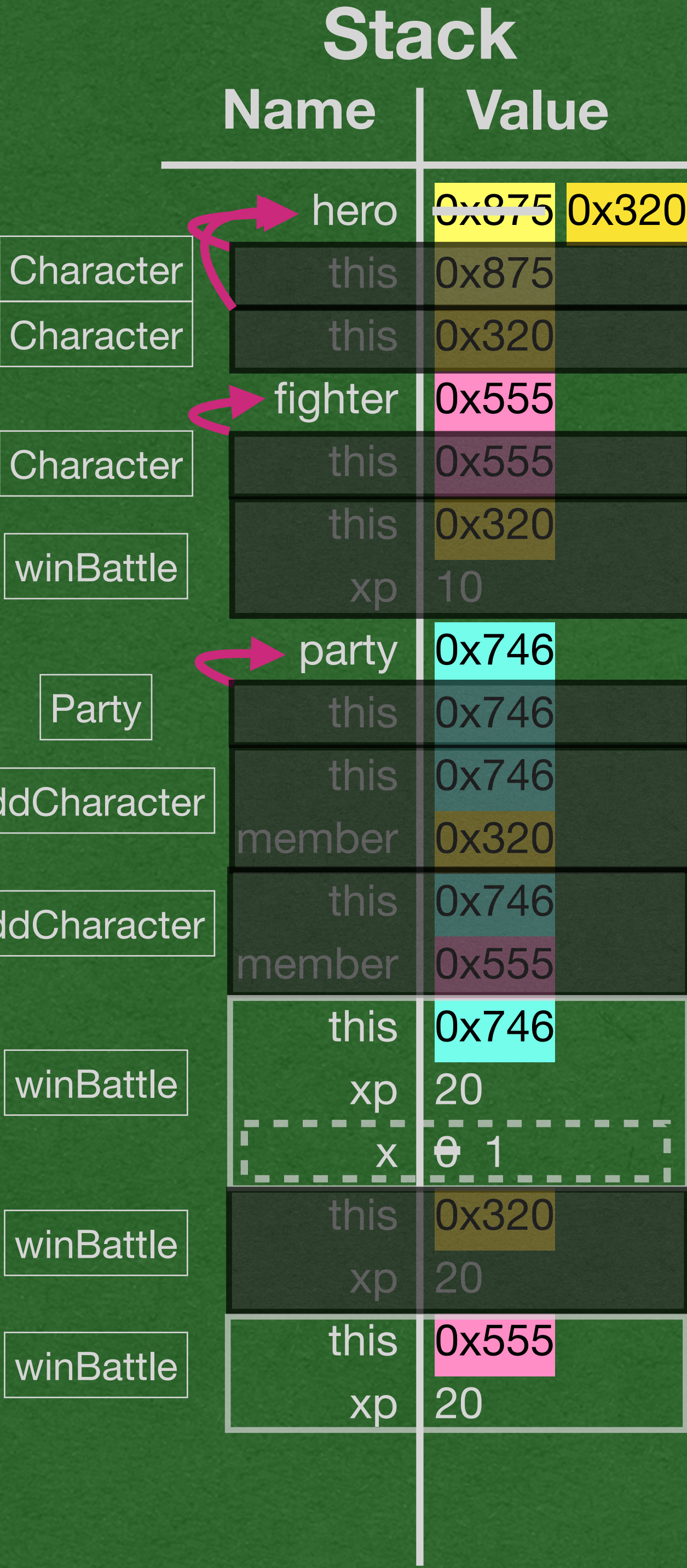
- Since we were at 0x320 in memory when this method is called
- 0x320 is the calling object
- *this* stores 0x320


```
public class Character {
    private int battlesWon = 0;
    private int expPoints = 0;

    public Character() {}
    public void winBattle(int xp) {
        this.battlesWon++;
        this.expPoints += xp;
    }
}
```

```
public class Party {
    private ArrayList<Character> members;
    private int battlesWon = 0;

    public Party() {
        this.members = new ArrayList<>();
    }
    public void addCharacter(Character member) {
        this.members.add(member);
    }
    public void winBattle(int xp) {
        this.battlesWon++;
        for (int x=0; x < this.members.size(); x++) {
            this.members.get(x).winBattle(xp);
        }
    }
    public static void main(String[] args) {
        Character hero = new Character();
        hero = new Character();
        Character fighter = new Character();
        hero.winBattle(10);
        Party party = new Party();
        party.addCharacter(hero);
        party.addCharacter(fighter);
        party.winBattle(20);
    }
}
```



- Repeat the process with 0x555

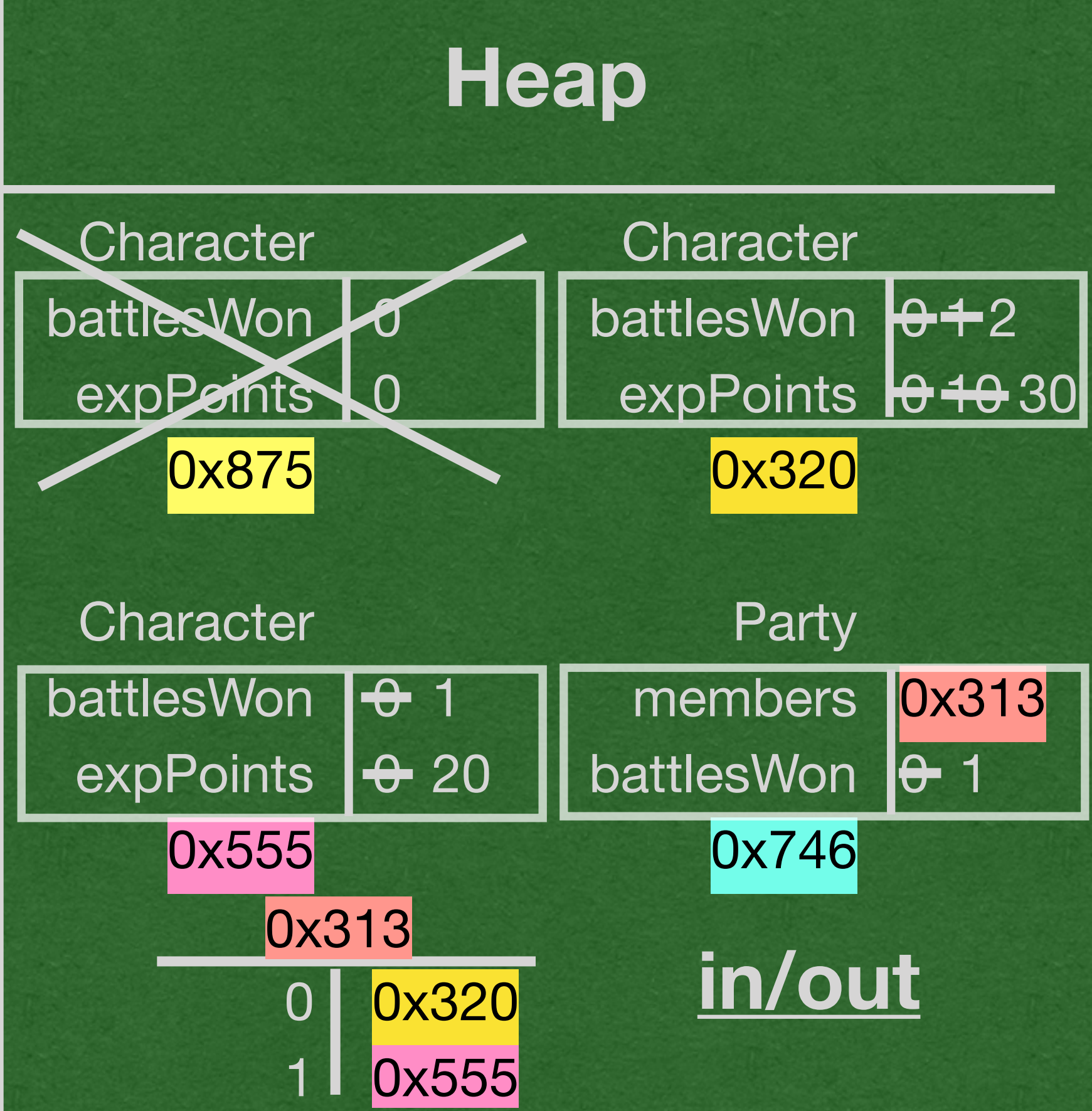
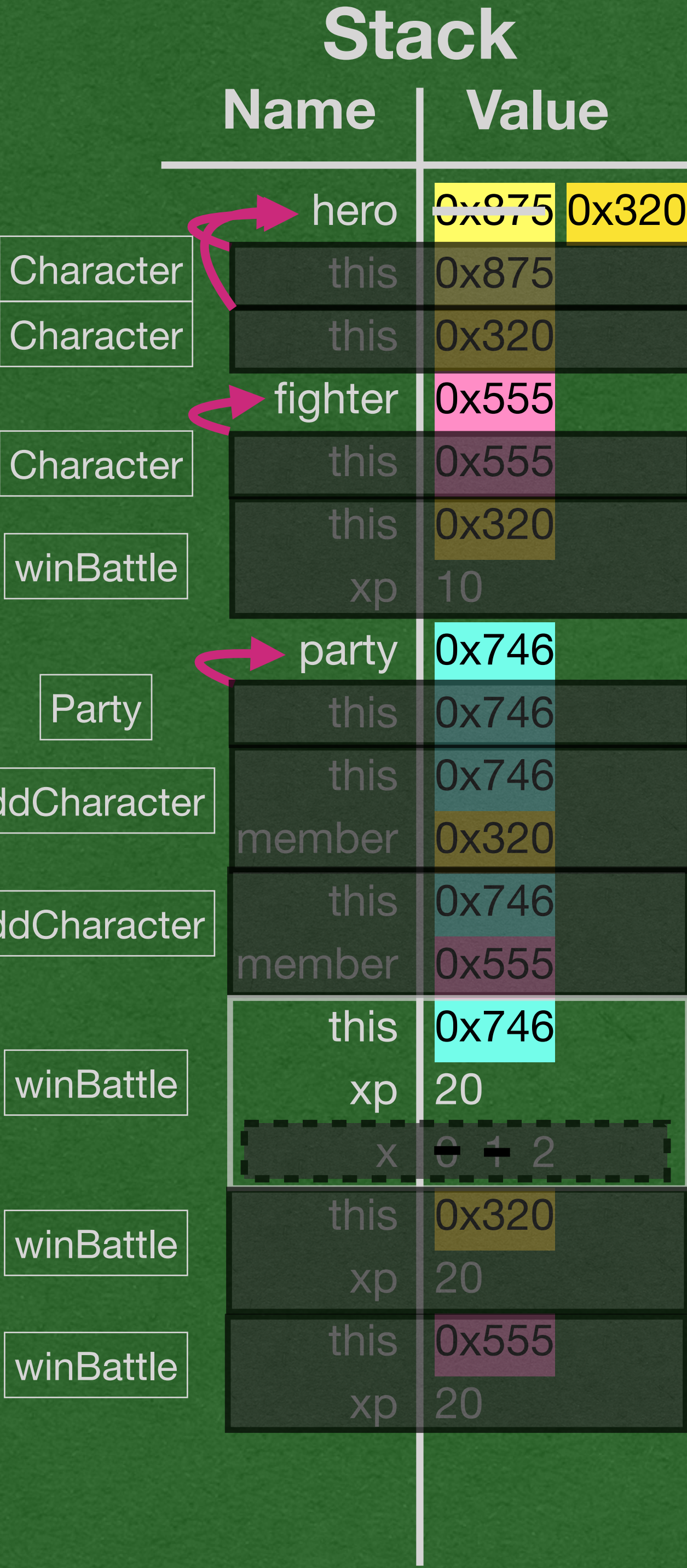
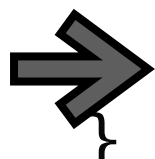
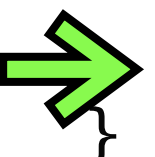

```
public class Character {
    private int battlesWon = 0;
    private int expPoints = 0;

    public Character() {}
    public void winBattle(int xp) {
        this.battlesWon++;
        this.expPoints += xp;
    }
}
```

```
public class Party {
    private ArrayList<Character> members;
    private int battlesWon = 0;

    public Party() {
        this.members = new ArrayList<>();
    }
    public void addCharacter(Character member) {
        this.members.add(member);
    }
    public void winBattle(int xp) {
        this.battlesWon++;
        for (int x=0; x < this.members.size(); x++) {
            this.members.get(x).winBattle(xp);
        }
    }

    public static void main(String[] args) {
        Character hero = new Character();
        hero = new Character();
        Character fighter = new Character();
        hero.winBattle(10);
        Party party = new Party();
        party.addCharacter(hero);
        party.addCharacter(fighter);
        party.winBattle(20);
    }
}
```



- The loop ends when x reaches 2
- x is removed from memory


```
public class Character {
    private int battlesWon = 0;
    private int expPoints = 0;

    public Character() {}
    public void winBattle(int xp) {
        this.battlesWon++;
        this.expPoints += xp;
    }
}

public class Party {
    private ArrayList<Character> members;
    private int battlesWon = 0;

    public Party() {
        this.members = new ArrayList<>();
    }
    public void addCharacter(Character member) {
        this.members.add(member);
    }
    public void winBattle(int xp) {
        this.battlesWon++;
        for (int x=0; x < this.members.size(); x++) {
            this.members.get(x).winBattle(xp);
        }
    }
    public static void main(String[] args) {
        Character hero = new Character();
        hero = new Character();
        Character fighter = new Character();
        hero.winBattle(10);
        Party party = new Party();
        party.addCharacter(hero);
        party.addCharacter(fighter);
        party.winBattle(20);
    }
}
```

